

# 基于矩阵变换和可调节环的部分重复码构造<sup>①</sup>



沈克勤, 孙 伟, 何亚锦, 张鑫楠

(长安大学 信息工程学院, 西安 710064)

通讯作者: 沈克勤, E-mail: [17862001617@163.com](mailto:17862001617@163.com)

**摘 要:** 目前在构造部分重复码 (Fractional Repetition Codes, FRC) 的研究方法中发现, 大多数是基于同构的分布式存储系统, 但实际的存储系统往往需要满足异构的特性. 为此, 本文提出了两种构造异构 FRC 的方法, 一种是基于矩阵变换构造的异构 FRC, 该方法用于构造重复度为 2, 节点存储容量异构的 FRC, 相比用正则图构造的同构 FRC, 具有算法计算复杂度低, 更符合现实存储系统的优点; 另外, 本文还提出了运用可调节环构造 FRC 的方法, 用于构造重复度为 2 或 3 的 FRC, 即可得到节点存储容量同构的 FRC 也可得到异构的 FRC. 与现有的 FRC 对比分析, 发现本文构造的 FRC 在节点存储容量上具有异构的特点, 修复局部性好, 同时构造算法运算复杂度低, 可以大范围的选择参数, 构造结构简单直观.

**关键词:** 分布式存储系统; 部分重复码; 矩阵变换; 环; 节点修复

引用格式: 沈克勤, 孙伟, 何亚锦, 张鑫楠. 基于矩阵变换和可调节环的部分重复码构造. 计算机系统应用, 2020, 29(12): 187-193. <http://www.c-s-a.org.cn/1003-3254/7713.html>

## Construction of Fractional Repetition Codes Based on Matrix Transformation and Adjustable Ring

SHEN Ke-Qin, SUN Wei, HE Ya-Jin, ZHANG Xin-Nan

(School of Information Engineering, Chang'an University, Xi'an 710064, China)

**Abstract:** According to the current method of constructing Fractional Repetition Codes (FRC), it is found that most of them are distributed storage systems based on isomorphism, but the actual storage systems often need to satisfy the characteristics of heterogeneity. To this end, this study proposes two methods for constructing heterogeneous FRC. One is a heterogeneous FRC constructed based on matrix transformation. This method is used to construct an FRC with a repeatability of 2 and a heterogeneous storage capacity of nodes. Compared with the existing isomorphic FRC constructed by regular graphs, it has the advantages of lower computational complexity and more in line with the real storage system. In addition, this study also proposes a method of constructing FRC using adjustable rings, which FRC is constructed with a repeatability of 2 or 3, which can obtain the FRC of the node storage capacity is isomorphic, and can also get the heterogeneous FRC. Compared with the existing FRC, it is found that the FRC constructed in this study has heterogeneous characteristics in node storage capacity, good repair locality, and low computational complexity of the construction algorithm. It can select parameters in a wide range and the construction structure is simple and intuitive.

**Key words:** distributed storage system; fractional repetition codes; matrix transformation; ring; node repair

随着大数据时代的到来, 数据资源呈现出快速增长的趋势, 数据的存储容量也随之不断增加. 传统的数据

存储系统已经不能适应当前海量数据存储, 分布式存储系统逐渐成为主流存储方式. 通过将海量数据分散的存

① 基金项目: 陕西省自然科学基金 (2019JM-386)

Foundation item: Natural Science Foundation of Shaanxi Province (2019JM-386)

收稿时间: 2020-05-03; 修改时间: 2020-05-28, 2020-06-10; 采用时间: 2020-06-15; csa 在线出版时间: 2020-11-30

储在多台互相独立物理设备上,分布式存储系统不仅很好的分担了存储负载,而且成本低廉,可扩展性能好,但是分布式存储系统中的这些物理存储设备容易发生故障,可造成大量数据丢失.因此,如何提高数据存储的可靠性就成为了分布式存储亟需解决的问题<sup>[1-3]</sup>.

为保证数据存储时的高可靠性和高可用性,传统的分布式存储系统中生成冗余数据的策略通常有“复制”和“纠删码”策略<sup>[4,5]</sup>.谷歌文件系统和 Hadoop 系统运用了三副本复制策略,将原始数据块复制成三个副本然后存储在系统中来保证存储的可靠性,这样会导致存储开销过大;为了减小存储开销,在实际系统中引入纠删码的冗余策略,但该策略在修复故障节点时会带来巨大的带宽开销.针对上述问题,Dimakis 等将网络编码的思想运用到分布式存储中,提出了再生码的概念<sup>[6]</sup>,有效减少了存储开销和修复带宽开销.目前对再生码的研究表明,主要表现在存储和带宽均衡曲线上的两个极值点,一个极值点对应最小存储再生码 MSRC (Minimum Storage Regenerating Code),另一个极值点对应最小带宽再生码 MBRC (Minimum Bandwidth Regenerating Code).文献 [7-9] 给出了一些好的再生码的构造方法.

但是,再生码的缺陷在于,在进行故障节点修复时,需要大量基于有限域上的计算,计算复杂度高,修复局部性复杂.为解决上述问题,El Rouayheb 和 Ramchandram 在 MBRC 的研究基础上提出了一种新型码——部分重复码 (Fractional Repetition Codes, FRC)<sup>[10]</sup>,该码可以进行精确无编码有效的修复.一般意义上的 FRC 由两部分组成:外部的编码是最大距离可分码 (Maximum Distance Separable, MDS) 和内部是重复码,该码修复故障节点无需任何编码操作,可以很好地降低故障修复时所需的带宽和磁盘 I/O 开销.目前对 FRC 的研究主要有基于组合设计构造的 FRC<sup>[11]</sup>,基于图构造的 FRC<sup>[12]</sup>,基于偏序集构造的 FRC<sup>[13]</sup>,基于二分图构造的局部修复的 FRC<sup>[14]</sup>,这些构造算法复杂,并且大多只能构造同构的 FRC,不能得到异构的 FRC.

为此,本文提出了两种构造方法,一种是基于矩阵变换构造的异构 FRC,该构造用于构造重复度为 2,节点存储容量异构的 FRC,该方法计算复杂度低,只需进行简单的异或运算就可得到节点存储容量异构的 FRC,相比现有的运用正则图构造的同构 FRC,该构造在节点存储容量上更符合现实的存储系统;另外,本文还提

出了运用可调节环构造的 FRC,该方法根据一定的存放规则能得到不同重复度的 FRC,主要构造重复度的情况,因为大部分对部分重复码的研究中重复度都是 2 或 3,同时该方法也可灵活的调节节点存储容量,即可得到节点存储容量同构的 FRC 也可得到节点存储容量异构的 FRC,可大范围选择参数,构造结构简单直观.同时本文最大的应用价值在于能无编码的修复节点存储容量异构的分布式存储系统中的故障节点,应用前景好,具有很好的实用价值.

## 1 基础知识

目前研究表明,对 MDS 码的研究已经十分成熟了,各种参数的 MDS 码都可得到.所以对部分重复码的研究主要体现在内部重复码的构造上.FRC 实际上是复制倍数为  $\rho$  的  $\theta$  个数据块在节点上的一种排列组合,复制生成的数据块都分别存储在不同的系统节点上.内部的重复码可用  $(n, k, d, \theta, \rho, \alpha)$  FRC 表示,其中  $n$  表示存储系统的节点数,  $\theta$  表示存储在节点中的数据块个数,  $\rho$  表示数据块的复制次数,  $\alpha$  表示每个节点的存储容量,  $d$  表示修复一个失效节点需连接的存活节点数,一般认为  $\alpha = d$ .数学上的定义如下:

定义 1<sup>[15]</sup>. 参数为  $(n, k, d)$  分布式存储系统的部分重复码  $C = (M, U)$ , 复制倍数为  $\rho$ , 是指特定的  $n$  个子集的集合  $U = \{U_1, U_2, \dots, U_n\}$ , 每个子集的元素均来自于符号集  $M = \{1, 2, \dots, \theta\}$ . 并且节点存储容量同构的 FRC 还需要满足下面条件:

- 1) 每个子集的大小均为  $d$ ;
- 2)  $M$  中的每一个元素都属于  $U$  中的子集, 每个子集数大小为  $\rho$ ;
- 3) 同构的 FRC 满足  $n\alpha = \rho\theta$ .

定义 2<sup>[16]</sup>.  $(d_1, d_2, \dots, d_m)$  正则图  $G(V, E)$  是一个无向图, 其中  $|V| = n$ ,  $V_1, V_2, \dots, V_m \subset V$ , 并且  $\sum_{i=1}^{\theta} |V_i| = n$ ,  $V_i \cap V_j = \emptyset$ . 顶点  $V_i$  的度为  $d_i (1 \leq i \leq m)$ , 若  $G(V, E)$  所有顶点的度都等于  $d$ , 则该  $G(V, E)$  叫作  $d$ -正则图, 若  $G(V, E)$  顶点的度不相等分别为  $d_1, d_2, \dots, d_m$ , 则称该  $G(V, E)$  为  $(d_1, d_2, \dots, d_m)$ -正则图, 也叫部分正则图.

## 2 基于矩阵变换的异构部分重复码构造

本节运用矩阵变换的思想, 结合部分正则图提出了一种新的节点存储容量异构的部分重复码, 相比文

献 [10] 和文献 [16] 中运用正则图和部分正则图构造的部分重复码, 本构造能得到节点存储容量更加多样的 FRC, 和传统 RS 码相比, 在修复单节点故障时, 修复局部性更好, 修复复杂度更优, 无需任何编码操作, 计算复杂度低. 具体构造算法如下:

该构造主要用于构造节点存储容量异构的 FRC, 适用于分布式存储系统节点数  $n$  为奇数的情况, 且构造的 FRC 中数据块的重复度  $\rho$  等于 2; 具体步骤如下:

步骤 1. 定义一个  $n$  阶的二进制循环置换矩阵  $C_n(d-1)$ , 其中  $n$  代表节点数,  $d-1$  表示每个节点存储容量同时也表示矩阵中每一行 1 的个数, 且需满足的条件为  $d > 3$ ,  $d$  为奇数; 同时我们设定  $C_n(d-1)$  矩阵的第一行在数学上满足的表达式为:  $c(t) = t + t^2 + \dots + t^{(d-1)/2} + t^{n-(d-1)/2} + \dots + t^{n-1}$ .

在矩阵的第一行确定后, 矩阵后面的每一行依次向右移动一位, 共移动  $n-1$  次, 最后生成  $C_n(d-1)$  矩阵;

步骤 2. 为得到节点存储容量异构的 FRC, 在步骤 1 的基础上引入矩阵  $S_n$  去调节步骤 1 中的  $C_n(d-1)$  矩阵,  $S_n$  矩阵生成方法为: 在  $(n-1)$  阶副对角线都为 1, 其他元素全为 0 的方阵后面加一行 0 和 0, 生成  $n \times n$  阶的  $S_n$  矩阵;

步骤 3. 将步骤 1 中的矩阵  $C_n(d-1)$  和步骤 2 中的矩阵  $S_n$  进行模 2 运算, 得到二进制矩阵  $P = C_n(d-1) + S_n \pmod{2}$ , 矩阵  $P$  和部分正则图存在相关联的关系, 用  $P = (m_{ij})_{n \times n}$ ,  $1 \leq i, j \leq n$  表示部分正则图的关联矩阵, 关联规则如下:

$$m_{ij} = \begin{cases} 1, & \text{图中第 } i \text{ 顶点与第 } j \text{ 个顶点相连} \\ 0, & \text{顶点 } i \text{ 和 } j \text{ 不相连} \end{cases}$$

由上面关系可知, 部分正则图的每一个顶点的度和矩阵的每一行中 1 的个数是相等的, 经过算法的验证, 发现矩阵  $P$  的不同行中会出现有  $d, d-1, d-2$  个 1 的情况, 因此对应的部分正则图的度有  $d, d-1, d-2$  三种情况, 记作  $(d, d-1, d-2)$ -部分正则图, 也就对应着构造的 FRC 的节点存储容量有  $d, d-1, d-2$  三种情况.

对构造的 FRC 的故障节点修复进行分析可知, 因为该 FRC 的重复度  $\rho = 2$ , 所以本构造能容忍单个节点出现故障, 又由于异构 FRC 的节点容量有  $d, d-1, d-2$  三种情况, 所以分以下 3 种情况讨论:

1) 若存储容量为  $d$  的节点出现故障, 那么只需要从另外的  $d$  个节点分别下载一个数据块即可直接修复;

2) 若存储容量为  $d-1$  的节点出现故障, 那么只需

要从另外的  $d-1$  个节点分别下载一个数据块即可直接修复;

3) 若存储容量为  $d-2$  的节点出现故障, 那么只需要从另外的  $d-2$  个节点分别下载一个数据块即可直接修复.

当系统中出现故障节点, 只需直接从其他存活的节点下载数据块修复, 修复选择性高, 无编码操作, 计算复杂度低. 根据上述构造算法给出如下具体实例.

例 1. 给定  $n = 7, d = 5$ , 根据构造方法步骤 1 得到矩阵  $C_7(4)$ , 其中  $C_7(4)$  是一个  $7 \times 7$  的二进制矩阵且第一行表示为  $c(t) = t + t^2 + t^5 + t^6$ , 第一行确定后, 后面的每一行依次向右移动一位, 最后生成  $C_7(4)$  矩阵, 如下所示:

$$C_7(4) = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

进一步运用矩阵  $S_7$  调节矩阵  $C_7(4)$ ,  $S_7$  矩阵是在 6 阶副对角线都为 1, 其他元素全为 0 的矩阵后面加一行 0 和 0 生成的, 如下所示:

$$S_7 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

得到  $S_7$  矩阵后, 通过  $P = C_7(4) + S_7 \pmod{2}$  算得矩阵  $P$ , 如下所示:

$$P = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

根据矩阵  $P$  能得到  $(3, 4, 5)$ -部分正则图, 即部分正则图的度有 5, 4, 3 这三种情况, 也就对应节点存储容量有 5, 4, 3 三种情况, 如图 1 所示.

若节点  $U_1$  发生故障, 需连接  $U_2, U_3, U_7$  这 3 个节点进行修复, 即从  $U_2, U_3, U_7$  这 3 个节点下载 1, 6, 7

数据块进行修复,修复过程如图2所示.同理,其他节点发生故障也可用相同的方法进行修复.

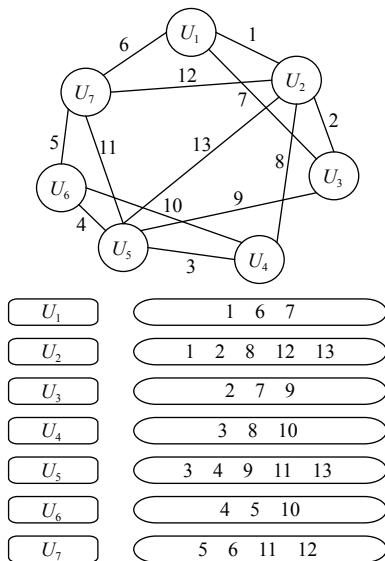


图1 (3, 4, 5)-部分正则图和对应 FRC 数据块存储结构图

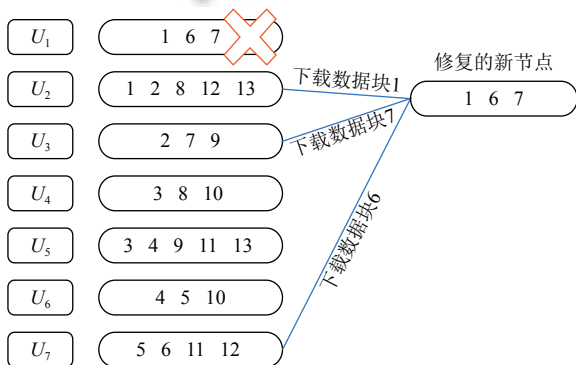


图2 故障节点修复图

### 3 基于可调节环的 FRC 构造

本节运用可调节环结构构造 FRC, 根据一定的存放规则去调节重复度的大小和节点存储容量, 规则是将数据元素放入相邻的节点所在环的边之间, 规定当每一个数据块依次放在两个相邻的节点之间时, 此时 FRC 的重复度为  $\rho = 2$ ; 当每一个数据块都放在 3 个相邻的节点之间, 此时 FRC 的重复度为  $\rho = 3$ , 同理可用相同的存放规则去调节重复度. 由同构 FRC 参数满足的条件  $n\alpha = \rho\theta$  可知, 当给定的参数满足该条件时, 可得到同构的 FRC, 若该等式不成立, 则可用可调节环构造节点存储容量异构的 FRC, 根据已有的对 FRC 的研究中发现, 大部分只考虑重复度为 2, 3 的情况, 具体构造算法如下.

#### 3.1 用可调节环去构造重复度 $\rho = 2$ 的 FRC

假设系统中节点用  $U_1, U_2, \dots, U_n$  表示, 节点中的数据块用  $\theta$  表示, 且  $[\theta] = \{1, 2, \dots, \theta\}$ , 将数据块按一定规则放入环中, 即从节点  $U_1$  开始, 将数据块 1 放在  $U_1$  和  $U_2$  所在的边上, 将数据块 2 放在  $U_2$  和  $U_3$  所在的边上, 数据块 3 放到  $U_3$  和  $U_4$  所在边上, 以此类推, 直到将  $\theta$  个数据块放完为止.

根据上述算法可以得到重复度  $\rho = 2$  的 FRC. 因为每一个数据块存在于相邻的 2 个节点所在环的边上, 每个数据块都会被两个节点所共有, 即得到的是重复度  $\rho = 2$  的 FRC. 若所给参数满足  $n\alpha = \rho\theta$ , 用可调节环可以构造节点存储容量同构的 FRC, 否则用可调节环可以构造节点存储容量异构的 FRC. 具体实例如下, 其中, 例 2 给定的是用可调节环构造的重复度  $\rho = 2$  的同构 FRC, 例 3 给定的是用可调节环构造的重复度  $\rho = 2$  的异构 FRC.

例 2. 给定  $n = 6, \theta = 12$ , 用可调节环去构造 FRC, 环结构和节点存储结构, 如图 3 所示.

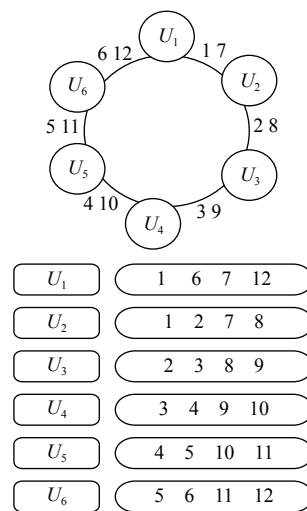


图3 可调节环结构和节点存储结构

由节点存储结构图可知该 FRC 满足  $n\alpha = \rho\theta$ , 是同构的 FRC, 重复度  $\rho = 2$ , 节点存储容量为  $\alpha = 4$ , 若节点  $U_1$  故障, 需要从  $U_2$  下载数块 1, 7, 从  $U_6$  下载数据块 6 和 12 修复  $U_1$ , 其他节点故障也可用相同的修复方式进行修复, 无需编码操作.

例 3. 给定  $n = 8, \theta = 21$ , 用可调节环去构造 FRC, 环结构和节点存储结构图, 如图 4.

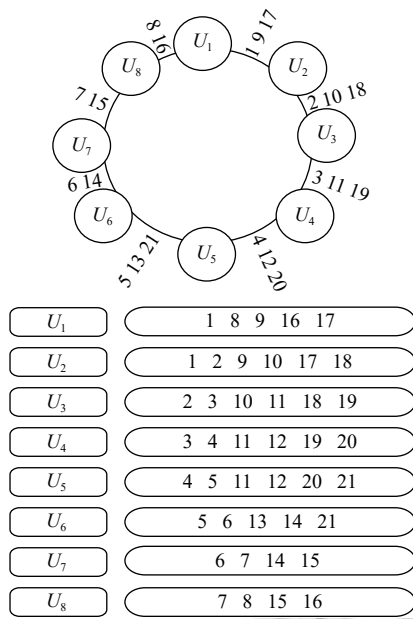


图4 可调节环结构和节点存储结构图

由节点存储结构图可知,该FRC不满足 $n\alpha = \rho\theta$ ,是异构的FRC,且重复度 $\rho = 2$ ,节点存储容量有4,5,6三种情况,可修复单节点故障,修复方式是直接从存活节点下载相应数据块进行修复。

### 3.2 用可调节环去构造重复度 $\rho = 3$ 的FRC

若分布式存储系统的节点用 $U_1, U_2, \dots, U_n$ 表示, $\theta$ 表示存储在节点中的数据块,且 $[\theta] = \{1, 2, \dots, \theta\}$ ,将 $\theta$ 个数据块按一定的规则放入环中,即从 $U_1$ 节点开始,将数据块1分别放到 $U_1U_2$ 和 $U_2U_3$ 所在的边上,将数据块2分别放到 $U_2U_3$ 和 $U_3U_4$ 所在边上,数据块3放到 $U_3U_4$ 和 $U_4U_5$ 所在边上,以此类推,直到将 $\theta$ 个数据块放完为止。

根据上述算法可得到重复度 $\rho = 3$ 的FRC.因为每一个数据块存在于相邻的3个节点所在的环之间,即每个数据块都会被3个节点共有,则得到的是重复度 $\rho = 3$ 的FRC.若所给参数满足 $n\alpha = \rho\theta$ ,用可调节环可以构造节点存储容量同构的FRC,否则可以构造节点存储容量异构的FRC.具体实例如下,其中,例4给定的是用可调节环构造的重复度 $\rho = 3$ 的同构FRC,如图5所示,例5给定的是用可调节环构造的重复度 $\rho = 3$ 的异构FRC,如图6所示。

例4. 给定 $\theta = n = 4$ ,则用可调节环去构造FRC,结构如下。

由上面的可调节环结构图和节点存储图可知,构造得到的码是重复度 $\rho = 3$ ,节点存储容量为3的同构

FRC.该FRC的故障节点修复方式为,当 $U_1$ 发生故障,可以直接重 $U_3$ 中下载1,3两个数据块,再从 $U_2$ 或 $U_4$ 中下载4这个数据块;当 $U_1$ 和 $U_2$ 同时发生故障时,可以直接从 $U_3$ 和 $U_4$ 节点中下载数据块进行修复,修复方式简单,无需任何编码。

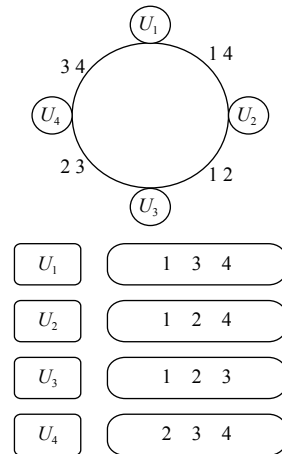


图5 可调节环结构和节点存储结构图

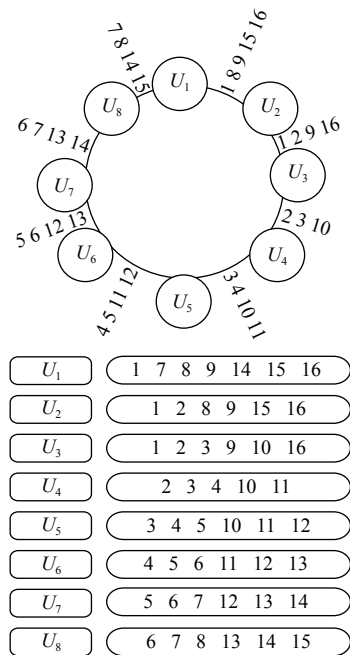


图6 可调节环结构和节点存储结构图

例5. 给定 $\theta = 16, n = 8$ ,用可调节环去构造FRC,结构如下。

由上面的可调节环结构图和节点存储图可知,构造得到的FRC是重复度 $\rho = 3$ ,节点存储容量异构的FRC,节点容量出现7,6,5三种情况.该FRC的故障节点修复方式为,直接从存活节点下载数据块,可最多修

复两个故障节点.

### 4 性能分析

对本文提出的两种新的构造进行性能分析, 主要与现有的 FRC 对比分析, 发现本文构造的 FRC 在节点存储容量上具有异构的特点, 修复局部性好, 同时在构造算法运算复杂度低, 可以大范围的选择参数, 构造结构简单直观.

#### 4.1 节点存储容量对比分析

对矩阵变换构造的异构 FRC 和已有的用正则图和部分正则图构造的 FRC 进行对比分析, 主要分析节点存储容量, 如表 1.

表 1 节点存储容量对比分析

节点数 $n$	正则图构造的 FRC 节点存储容量 $d$	部分正则图构造的 FRC 节点存储容量和 $d$	基于矩阵变换的异构 FRC 节点存储容量 $d$
7	5	5, 4	5, 4, 3
9	7	7, 6	7, 6, 5
11	9	9, 8	9, 8, 7
13	11	11, 10	11, 10, 9

表 1 只列举了部分情况, 可以发现本文提出的基于矩阵构造的异构 FRC 相比于正则图构造的 FRC 在节点存储容量上是异构的, 并且本文提出的构造方法在节点修复选择度上更优.

对基于可调节环构造的 FRC 进行对比分析, 相比于文献 [10] 提出的运用正则图构造的 FRC 本构造在重复度上的选择性更灵活, 正则图只能构造  $\rho = 2$  的同构 FRC, 用可环结构可以得到重复度多样的同构或异构的 FRC, 构造算法更简单直观.

#### 4.2 参数选择对比分析

根据已有研究表明, 大多数构造 FRC 的方法都对参数有明显的限制, 对比分析得本文提出的基于可调节环构造的 FRC, 在参数选择上更具有灵活性, 对比分析结果, 如表 2. 分析结果. 表 2 中各参数含义解释如下:  $\alpha$  是 FRC 的节点存储容量,  $d$  表示修复单个节点时需要连接的节点数, 一般意义上  $\alpha = d$ ,  $\rho$  表示 FRC 的数据重复度,  $\theta$  表示系统中数据块,  $n$  表示系统中的节点数,  $q$  是素数,  $h$  是 Hadamard 矩阵的阶数.

#### 4.3 修复局部性对比分析

修复局部性是指在修复故障节点时需要连接的存活节点数. 当单节点出现故障时, 运用正则图构造的 FRC 需要连接的节点数为  $d$ , 即修复局部性为  $d$ , 运用基于矩阵变换构造的异构 FRC 需要连接的节点数有

$d, d-1, d-2$  三种情况, 即修复局部性为  $d, d-1, d-2$  三种情况, 修复局部性更好. 另外, 当出现单节点故障时,  $(n, k)$ RS 码需要连接  $k$  个节点先恢复原始文件来修复出现故障的节点, 修复局部性为  $k$ ; 基于矩阵变换构造的异构 FRC 需要连接的节点数有  $d, d-1, d-2$  三种情况, 修复局部性为  $d, d-1, d-2$  三种情况, 又由于研究的 FRC 都是  $d < k$ , 所以可知和  $(n, k)$ RS 对比, 基于矩阵变换构造的异构 FRC 具有更好的修复局部性.

表 2 不同构造方法参数对比分析图

构造方法	$\alpha = d$	$\rho$	$\theta$	$n$
欧拉几何构造的 FRC <sup>[17]</sup>	$2 \leq \alpha \leq q$	$2 \leq \alpha \leq K$	$\alpha(q^m - 1)$	$\rho(q^m - 1)$
RS 码构造的 FRC <sup>[18]</sup>	$2 \leq \alpha \leq q$	$2 \leq \rho \leq q$	$\alpha q$	$\rho q$
对称设计 <sup>[19]</sup>	$q+1$	$q+1$	$q^2+q+1$	$q^2+q+1$
Hadamard 设计 <sup>[11]</sup>	$2h$	$2 \leq \rho \leq 4h$	$4h$	$2\rho$
基于矩阵变换设计	$d, d-1, d-2$	$\rho = 2$	没有明确限制	$n$ 为奇数
基于可调节环设计	没有明确限制	$\rho \geq 2$	没有明确限制	没有明确限制

图 7 给定的实例是基于矩阵变换构造的异构 FRC 和  $(n, k)$ RS 码的在修复局部性方面的对比情况, 当修复节点存储容量为  $d (d < k)$  的节点时, 可知基于矩阵变换构造的异构 FRC 的修复局部性恒为  $d (d < k)$ , 但 RS 码的修复局部性与  $k$  (正整数) 是一次线性关系.

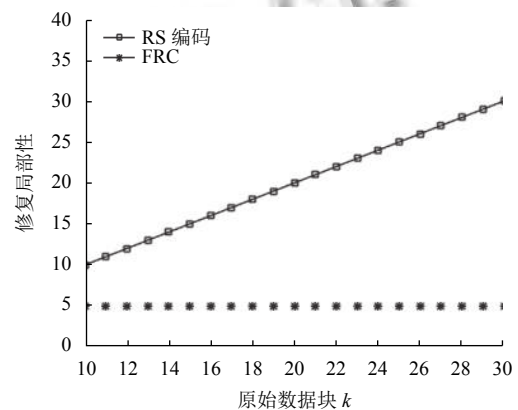


图 7 修复局部性与数据块  $k$  的关系图

#### 4.4 构造算法运算复杂度

衡量一个算法的优劣, 通常需要考虑算法构造时涉及的运算复杂度, 即将算法写成程序在实际的计算机系统中运行时, 涉及的计算量. 本文基于矩阵变换的异构 FRC, 由构造算法可知, 构造一个异构的 FRC 需要进行  $d-1$  次加法运算和  $n^2$  次模 2 加运算; 运用可调节环构造

的 FRC, 构造时不需要任何计算量, 只需在环上直接进行调节即可, 相比于基于矩阵变换的异构 FRC, 用可调节环得到的 FRC, 在构造算法运算复杂度表现的更优.

将基于矩阵变换的异构 FRC 和运用偏序集构造的 FRC<sup>[10]</sup> 在构造算法运算复杂度进行对比分析, 在文献 [10] 中, 构造算法时运用到了加法, 乘法运算和基于集合上的运算, 明显可知本文算法运算复杂度更低.

## 5 结论

考虑实际的分布式存储系统大多需要满足异构的特性. 为此, 本文提出了两种构造方法, 一种是基于矩阵变换构造的异构 FRC, 该构造主要用于构造重复度为 2, 节点存储容量异构的 FRC, 相比用正则图构造的同构 FRC, 该构造更符合现实的存储系统; 另外, 本文还提出了运用可调节环构造的 FRC, 构造得到了重复度为 2 或 3 的 FRC, 该方法即可得到节点存储容量同构的 FRC 也可得到节点存储容量异构的 FRC. 与现有的 FRC 对比分析, 发现本文构造的 FRC 在节点存储容量上具有异构的特点, 修复局部性好, 同时在构造算法运算复杂度低, 可以大范围的选择参数, 构造结构简单直观. 将来如何去构造更多样的异构 FRC 是研究的热点.

## 参考文献

- 1 Wang YJ, Xu FL, Pei XQ. Research on erasure code-based fault-tolerant technology for distributed storage. *Chinese Journal of Computers*, 2017, 40(1): 236–255.
- 2 Li CD, Zhou ZH, Zhai XP. On a class of multi-source distributed storage with exact repair. *IEEE Access*, 2018, 6: 20704–20711. [doi: [10.1109/ACCESS.2018.2825324](https://doi.org/10.1109/ACCESS.2018.2825324)]
- 3 Cai L, Zhu YY. The challenges of data quality and data quality assessment in the big data era. *Data Science Journal*, 2015, 14: 2. [doi: [10.5334/dsj-2015-002](https://doi.org/10.5334/dsj-2015-002)]
- 4 Yuan ZM, Liu HY. Efficiently coding replicas to erasure coded blocks in distributed storage systems. *IEEE Communications Letters*, 2017, 21(9): 1897–1900. [doi: [10.1109/LCOMM.2017.2709312](https://doi.org/10.1109/LCOMM.2017.2709312)]
- 5 Rizzo L. Effective erasure codes for reliable computer communication protocols. *ACM SIGCOMM Computer Communication Review*, 1997, 27(2): 24–36. [doi: [10.1145/263876.263881](https://doi.org/10.1145/263876.263881)]
- 6 Dimakis AG, Godfrey PB, Wu YN, *et al.* Network coding for distributed storage systems. *IEEE Transactions on Information Theory*, 2010, 56(9): 4539–4551. [doi: [10.1109/TIT.2010.2054295](https://doi.org/10.1109/TIT.2010.2054295)]
- 7 Rashmi KV, Shah NB, Kumar PV. Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction. *IEEE Transactions on Information Theory*, 2011, 57(8): 5227–5239. [doi: [10.1109/TIT.2011.2159049](https://doi.org/10.1109/TIT.2011.2159049)]
- 8 Huang K, Parampalli U, Xian M. On secrecy capacity of minimum storage regenerating codes. *IEEE Transactions on Information Theory*, 2017, 63(3): 1510–1524. [doi: [10.1109/TIT.2016.2647601](https://doi.org/10.1109/TIT.2016.2647601)]
- 9 Mahdavian K, Khisti A, Mohajer S. Bandwidth adaptive & error resilient MBR exact repair regenerating codes. *IEEE Transactions on Information Theory*, 2019, 65(5): 2736–2759. [doi: [10.1109/TIT.2018.2878223](https://doi.org/10.1109/TIT.2018.2878223)]
- 10 El Rouayheb S, Ramchandran K. Fractional repetition codes for repair in distributed storage systems. *Proceedings of the 2010 48th Annual Allerton Conference on Communication, Control, and Computing*. Monticello, IL, USA. 2010. 1510–1517.
- 11 Olmez O, Ramamoorthy A. Fractional repetition codes with flexible repair from combinatorial designs. *IEEE Transactions on Information Theory*, 2016, 62(4): 1565–1591. [doi: [10.1109/TIT.2016.2531720](https://doi.org/10.1109/TIT.2016.2531720)]
- 12 Silberstein N, Etzion T. Optimal fractional repetition codes based on graphs and designs. *IEEE Transactions on Information Theory*, 2015, 61(8): 4164–4180. [doi: [10.1109/TIT.2015.2442231](https://doi.org/10.1109/TIT.2015.2442231)]
- 13 Aydinian H, Boche H. Fractional repetition codes based on partially ordered sets. *Proceedings of 2017 IEEE Information Theory Workshop*. Kaohsiung, China. 2017. 51–55.
- 14 Su YS. Optimal pliable fractional repetition codes that are locally recoverable: A bipartite graph approach. *IEEE Transactions on Information Theory*, 2019, 65(2): 985–999. [doi: [10.1109/TIT.2018.2876284](https://doi.org/10.1109/TIT.2018.2876284)]
- 15 朱兵, 李挥, 陈俊, 等. 基于可分组设计的部分重复码研究. *通信学报*, 2015, 36(2): 98–105.
- 16 Gupta MK, Agrawal A, Yadav D. On weak dress codes for cloud storage. <http://arxiv.org/abs/1302.3681>, 2013-02-15.
- 17 Tai YY, Lan L, Zeng LQ, *et al.* Algebraic construction of quasi-cyclic LDPC codes for the AWGN and erasure channels. *IEEE Transactions on Communications*, 2006, 54(10): 1765–1774. [doi: [10.1109/TCOMM.2006.881361](https://doi.org/10.1109/TCOMM.2006.881361)]
- 18 Su YS. On the construction of local parities for  $(r, t)$ -availability in distributed storage. *IEEE Transactions on Communications*, 2017, 65(6): 2332–2344. [doi: [10.1109/TCOMM.2017.2682861](https://doi.org/10.1109/TCOMM.2017.2682861)]
- 19 Zhu B, Li H. Adaptive fractional repetition codes for dynamic storage systems. *IEEE Communications Letters*, 2015, 19(12): 2078–2081. [doi: [10.1109/LCOMM.2015.2496197](https://doi.org/10.1109/LCOMM.2015.2496197)]